comprise at least one common register 10, two CPUs 12 and 22, two RAMs 14 and 24, two DMAs 16 and 26, two system data buses 17 and 27, and two FIFOs 18 and 28.

[0012]The CPUs 12 and 22 may have been de-coupled so that they may operate independently. Thus, the state of one CPU may not affect the state of the other. Each CPU 12 and 22 may have a separate memory space RAMs 14 and 24. Furthermore, CPUs 12 and 22 may not share any RAM. CPU 12 may access only RAM 14 and CPU 22 may only access RAM 24. Thus, there may be no conflicting attempts to access memory.

[0013]Multiple levels of processing may be required for data transmission or receipt. For example, a first set of processing may be performed by CPU 12 and a second set of processing may be performed by CPU 22. Thus, CPU 12 may need to send data that it may have processed to CPU 22. CPU 12 may instruct DMA 16 to transfer data located in RAM 14 to CPU 22. These instructions may include, for example, the data location and data length or data start and data end. The data may be copied to FIFO 18 under the control of DMA 16 on system data bus 17. DMA 16 may inform CPU 12 that the data has been sent.

[0014]DMA 26 may be instructed by CPU 22 to await a data transmission. DMA 26 may retrieve the data from FIFO 18 and may write it to RAM 24 using system data bus 27. DMA 26 may inform CPU 22 that all the data has been received. Other information may also be shared between CPUs 12 and 22. Such information may be stored in common register 10 by CPU 12 and read from register 10 by CPU 22 (or vice versa).

[0015] In data transmission from CPU 22 to CPU 12, the process is reversed. CPU 22 may instruct DMA 26 to transfer data located in RAM 24 to CPU 12. The data may be copied to FIFO 28 under the control of DMA 26 on system data bus 27. DMA 26 may inform CPU 22 that the data has been sent. DMA 16 may have been instructed to await a data transmission, may retrieve the data from FIFO 28, and may write it to RAM 14 using system data bus 17. DMA 16 may inform CPU 12 that data has been received.

[0016] It should be noted that, from the perspective of CPU 12, FIFO 18 may be write only and FIFO 28 may be read only. From the perspective of CPU 22, FIFO 18 may be read only and FIFO 28 may be write only.

[0017] Additional common registers, CPUs, RAMs, DMAs, system data buses, and FIFOs are also within the scope of this invention. Additionally, having only one FIFO, DMA, and/or system data bus is also within the scope of this invention.

[0018] Reference is now made to Fig. 2, a data flow illustration of data transmission from RAM 14 to RAM 24 and from RAM 24 to RAM 14, in accordance with an embodiment of the present invention. Data flow direction is indicated by the arrows. Elements described with respect to Fig. 1 are numbered similarly. DMAs 16 and 26 may each comprise at least two channels, of which one channel may be used for read operations and one channel may be used for write operations. Receive channels are referred to hereinbelow as Rx channel 19 and Rx channel 29 for DMA 16 and 26 respectively. Transmit channels are referred to hereinbelow as Tx channel 11 and Tx channel 21 for DMA 16 and 26 respectively.

[0019] Each side of FIFO 18/28 may be handled by a different DMA 16/26. For example, data input to FIFO 18 may be controlled by Tx channel 11, while data read

from FIFO 18 may be controlled by Rx channel 29. Data input to FIFO 28 may be controlled by Tx channel 21, while data read from FIFO 28 may be controlled by Rx channel 19. Thus, each FIFO 18/28 may control the flow of data on both sides, which may ensure that no underflow or overflow errors will occur. FIFO 18/28 may be as small as 2 - 4 words long, since correct data receipt and transmission may have been assured.

[0020] An exemplary embodiment of the system and method of the present invention may be described using a chip with two processing units that may implement the media access control (MAC) and the physical layer protocol (PHY) of a network communication system. Devices attached to a network may need to handle information transfer to and from the network. There are known standards for handling this communication. The PHY and MAC protocols implement different layers of the networking protocol used to access a network.

[0021] In an embodiment of the present invention, two CPUs are embedded on the same chip. Thus, CPU 12 may control the MAC protocol implementation, whereas CPU 22 may control the PHY protocol implementation. Such an implementation may include both receive and transmit functions. This example is given for clarity purposes only and does not limit the scope of the invention.

[0022] CPU 12 may be instructed by the system in which the chip is embedded to transfer data to the network. CPU 12 may perform initial data processing using the MAC protocol, for example, packetizing the data. CPU 12 may instruct DMA 16 to send the required data. CPU 12 may write information to common register 10, for example, status information, so that it may be used by CPU 22 if necessary. Tx channel 11 may retrieve the data from the correct location in RAM 14 and may write